

Semestrální práce z předmětu KIV/WEB

Lukáš Černý

A13B0286P

luccerny@students.zcu.cz

6. prosinec, 2014

Obsah

1	Úvod	3
1.1	Představení aplikace	3
1.2	Popis instalace	3
2	Popis aplikace	4
2.1	Půjčovna	4
2.1.1	Nepřihlášený uživatel	4
2.1.2	Přihlášený uživatel	4
2.2	Administrace	5
2.2.1	Uživatelé	5
2.2.2	Filmy	5
2.2.3	Statistika	5
2.3	Adresářová struktura	6
2.3.1	Root	6
2.3.2	Controllors	6
2.3.3	Models	7
2.3.4	Templates	8
2.4	Datový model	9
2.5	Použité technologie	10
3	Závěr	12
3.1	Zhodnocení práce	12

Jméno a příjmení: Lukáš Černý

Osobní číslo: A13B0286P

Kontaktní e-mail: luc.cerny@gmail.com

Orion login: luccerny

Obor/kombinace: INF

Rok studia: 2

Název práce: Půjčovna filmu

Návrh databáze by měl sloužit pro realizaci půjčovny filmů. Databáze obsahuje dvě hlavní tabulky. Tabulka PROFIL reprezentuje uživatele a podle jeho nastavených práv se zobrazuje obsah. Spojovací tabulka mezi PROFIL a FILM bude ukládat datum vypůjčení a slouží pro uchování historie. Druhá tabulka FILM slouží pro zobrazování obsahu a její spojené tabulky pro filtrování. Návrh databáze bych rád použil i v předmětu KIV/WEB.

Při řešení mého zadání chci použít (ponechte tu variantu, kterou chcete použít):

- SŘBD MySQL na svém hardware (PC, notebook či tablet)

Stručně slovně popište maximálně 2 vhodné dotazy (nepovinná část):

Vyplněný formulář zadání semestrální práce doručte e-mailem ve formátu PDF svému cvičícímu nejpozději do **2. listopadu 2014**. Nezapomeňte také poslat odpovídající návrh relačního modelu, taktéž ve formátu PDF.

Mezní termín odevzdání a předvedení semestrální práce je **6. únor 2015 do 12:00**. Tento termín platí též pro získání zápočtu.

Kapitola 1

Úvod

1.1 Představení aplikace

Webová aplikace simuluje půjčování filmů. Jelikož se nejedná o konkrétní zadání pro firmu, může být použita obecně. S doděláním některých modulů může sloužit pro půjčování filmů na fyzických nosičích nebo jako datový soubor s DRM ochranou.

1.2 Popis instalace

1. Nainstalujte si Apache, PHP, MySQL, phpMyAdmin. Můžete použít již hotové balíky např. (XAMP, WAMP).
2. V konfiguračním souboru PHP změňte AllowOverride z None na All a povolte externí moduly.
3. Stáhněte si ¹ projektu ze serveru GitHub do root složky pro php soubory.
4. V aplikaci phpMyAdmin ² spusťte SQL script³ pro vytvoření databáze.
5. Spusťte aplikaci v prohlížeči.

¹stažení přes git příkazem *git clone*

²doména/phpmyadmin

³přiložený ve složce SQL

Kapitola 2

Popis aplikace

2.1 Půjčovna

2.1.1 Nepřihlášený uživatel

Přijde-li zákazník na webové stránky, uvidí seznam filmů, u kterých je uvedený režisér, hlavní herec, rok vydání vydání, jazyk, ve kterém je film dabován a náhled obalu. Přes všechny tyto uvedené informace může uživatel fyltrovat obsah. Jedná-li se o nového zákazníka, který nemá vytvořený účet, může si ho vytvořit po kliknutí na tlačítko *registrovat*. Po vyplnění údajů se poté může přihlásit. Celá aplikace je ošetřená, aby se nepřihlášený uživatel nedostal na místa, kam se bez přihlášení nedostane. Při pokusu o navštívení takových stránek vyhodí aplikace chybovou stránku.

2.1.2 Přihlášený uživatel

Přihlášený uživatel získá možnost zapůjčení jednoho či více filmů. Možnosti filtrace zůstávají stejné. Přehled aktuálně vybraných filmů k zapůjčení je vidět po stisknutí na ikonu košíku, která se nachází vedle tlačítka na odhlášení. Zákazník zde vidí nejen aktuálně vybrané filmy, ale i celkovou cenu. Není-li s výběrem spokojený, je zde možnost film z košíku odebrat. Po pokračování v objednávce je náhled kontaktních informací a dokončení objednávky. Poté se zobrazí tlačítko pro zobrazení faktury. Kontaktní údaje a heslo má uživatel možnost si změnit po kliknutí na svoje jméno v horní části. Zobrazí se mu formulář a možností úpravy. Vidí zde také historii všech půjčených filmů.

2.2 Administrace

Do administrátorské sekce se administrátor dostane po zadání */admin* za doménu půjčovny. Po přihlášení má administrátor možnost spravovat obsah půjčovny.

2.2.1 Uživatelé

Po vyplnění správných přístupových údajů je uživatel přesměrován na tuhle stránku. Vidí zde seznam vytvořených účtů, a u každého z nich je možnost smazání a změny oprávnění. V horní liště jsou tlačítka pro zobrazení pouze administrátorů, běžných uživatelů nebo všech.

2.2.2 Filmy

Další záložkou je správa filmů. Zde je jediná možnost, kterou lze s filmem provést, a to je jeho smazání. V horní části je možnost přidání nového filmu. Nachází se zde formulář s připravenými okny pro název, rok vydání, ceny půjčoveného, nabídka pro výběr režiséra, herce, jazyka a výběr obrázku. Není-li v nabídkách požadovaná položka, nachází se za nabídkou okno pro vytvoření nové položky. Budete-li chtít přidat již existujícímu filmu další jazyk, stačí při vytvoření filmu zadat pouze název a jazyk.

2.2.3 Statistika

S poslední sekci se nachází statistika půjčování. U každého již půjčeného filmu je zobrazeno číslo, kolik lidí si film půjčilo.

2.3 Adresářová struktura

2.3.1 Root

index

Spouští celou aplikaci. Je zde *autoloader*, který se stará o načtení daného php souboru. Startuje *sessions*, nastavuje filtr zobrazení filmů a předává url směrovači pro další zpracování.

.htaccess

Soubor pro vytváření hezké url.

2.3.2 Controllers

Zajišťují správný běh aplikace. Směřují pohyb uživatele a vykreslují obsah.

AbsController

Abstraktní třída, kterou dědí všechny kontrolery. Obsahuje metody pro zobrazení šablony, nastavení url a kontroly přihlášení.

AdminController

Ověřuje přihlášení administrátora a slouží jako směrovač v administrátorském rozhraní.

AjaxController

Skript pro přidávání filmů do košíku.

BasketController

Obsluhuje vše spojené k košíkem. Zobrazení filmů i mazání.

BorrowController

Přidává záznam do databáze o dokončení objednávky.

InvoiceController

Sestavuje šablonu faktury z půjčených filmů a informací o uživateli.

LoginController

Ověřuje skrz databázi, zda uživatel zadal správné přihlašovací údaje. Pokud ano, přihlásí ho.

ProfilController

Spravuje profil uživatele při změně údajů a zobrazuje historii půjčených filmů.

RegistrationController

Ověřuje správnost zadaných informací a vytváří uživatele v databázi.

RouterController

Směruje celý provoz aplikace. Vytváří instance všech tříd a kontroluje url adresu.

TwigController

Třída podle návrhového vzoru jedináček. Vytváří Twig registraci a obsahuje metodu pro zpracování šablony.

2.3.3 Models

Pomocné třídy pro přípravu dat z/do databáze.

Admin_movies, Admin_statistics, Admin_users

Obsahuje metody pro zpracování dat prováděné v administrátorském rozhraní.

Database

Třída podle návrhového vzoru jedináček. Obsahuje metody pro přístup do databáze.

File

Script pro zpracování nahrávaného obrázku.

Homepage

Připravuje data pro vykreslení úvodní stránky, kde je vidět seznam všech filmů.

config.inc

Konfigurační soubor.

2.3.4 Templates

Šablony pro vykreslení obsahu řízené php soubory ve složce Controllers. Pro rozeznání šablon pro nepřihlášeného uživatele je do názvu přidána zkratka *_nlg*. Šablony využívají funkcí Twigu jako jsou cykly, podmínky a dědění jiné šablony.

2.4 Datový model

Databázový model je navržen tak, aby simuloval půjčování knih. V návrhu uvažuji pět hlavních entit (film, profil, režiser, herec, dabing). Pro vytvoření a naplnění databáze je přiložen SQL skript.

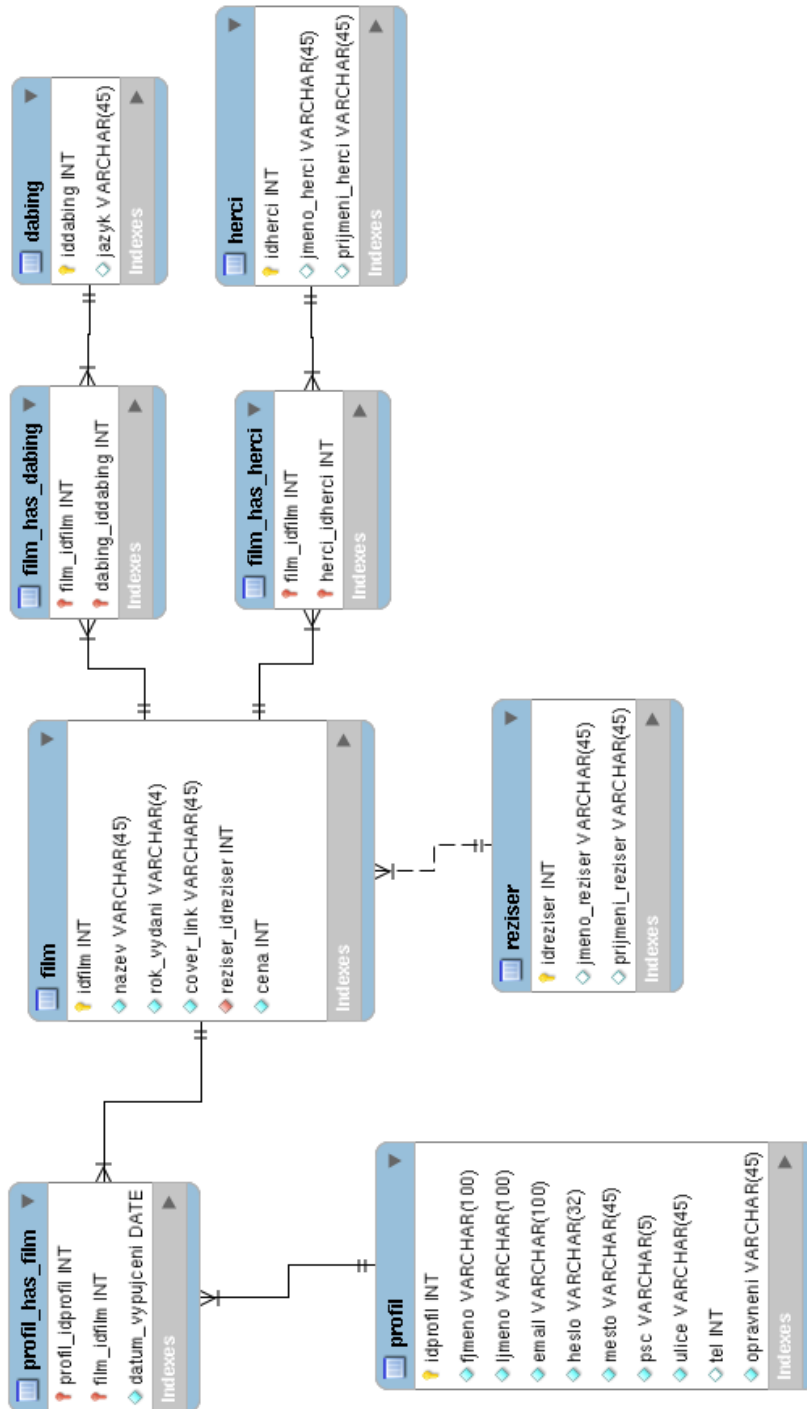
Pro výše uvedené entity jsem zvolil následující vztahy:

- *film-reziser* - Každý film může mít pouze jednoho režiséra, ovšem režisér může režírovat více filmů. Z toho důvodu jsem zvolil vazbu 1:N.
- *film-herci* - Ve filmu hrají herci, a těch je více než jeden. A herci hrají ve v více filmech. Abych dolíli tohoto propojení, musel jsem zvolit vazbu M:N. Propojovací tabulka nese název *film_has_herci*.
- *film-dabing* - V půjčovně nabízím filmy s různým jazykem dabování. Abych docílil toho, že film má více jazykových verzí, použil jsem vazbu M:N. Propojovací tabulka nese název *film_has_dabing*.
- *profil-film* - Pro záznam historie půjčování filmů u jednotlivých zákazníků jsem zvolil vztah M:N. Docílil jsem toho, že mám přehled, jaký zákazník ci půjčil jaký filmy a do rozkladové tabulky *profil_has_film* jsem přidal atribut *datum_vypujceni*.

Entity:

- *profil* - Všechny potřebné informace o registrovaném uživateli. Obsahuje atributy *fjmeno* pro záznam křestního jména, *fjmeno* pro příjmení, *email*, *heslo*, *mesto*, *psc*, *tel* a *opraveni*.
- *film* - Základní informací o filmu. Je zde uveden *nazev*, *rok_vydani*, *cover_link* pro uchování názvu obalu, *cena* a *reziser_idreziser*, kde je uloženo ID režiséra z tabulky *reziser*.
- *herci* - Uchovává jméno herce jako *jmeno_herci* a jeho příjmení jako *prijmeni_herci*.
- *dabing* - Tabulka je typu číselník, obsahuje pouze atribut *jazyk*.
- *reziser* - Je zde pro odstranění redundance. Obsahuje *jmeno_reziser* a *prijmeni_reziser*.
- *film_has_herci* - Rozkladová tabulka mezi *film* a *herci*.
- *film_has_dabing* - Rozkladová tabulka mezi *film* a *dabing*.

- *profil_has_film* - Rozkladová tabulka mezi *profil* a *film*. Dalším atributem je *datum_vypujceni*, který uchovává historii půjčování filmů uživatele.



2.5 Použité technologie

Aplikace je napsaná v programovacím jazyce PHP. Pro přehlednost a snadnou rozšiřitelnost jsem použil architekturu MVC. Písmeno M ve zkratce znamená *Model* a slouží pro získání dat z databáze. Data jsou následně zobrazována přes šablonovací systém Twig. Výhoda tohoto systému je v oddělení HTML od kódu psaném v programovacím jazyce. Posledním písmenem je C, které představuje Controller a zajišťuje chod celé aplikace. Předává data získané z databáze šablonám.

Databáze je postavená na databázovém systému MySQL. Připojení je přes ovladač PDO. Grafická vizualizace je možná přes aplikaci phpMyAdmin, a můžete ji použít pro vytvoření databáze z SQL skriptu.

Grafická stránka aplikace je napojena na framework Twitter Bootstrap. Pro půjčovnu jsem použil šablonu s názvem Jumbotron a mírně upravil. Administrativní rozhraní je vytvořené v šabloně SB Admin. Jelikož jsem zajistil veškeré možné pády aplikace a přesměrování na stránku *Error 404*, vytvořil jsem si vlastní chybovou stránku. Použitá šablona se jmenuje Ohh a je volně ke stažení.

Pro vkládání filmů do košíku jsem použil JavaScript. A to konkrétně tak, že při stisknutí tlačítka *Půjčit* se zavolá javascript funkce, která přes jQuery vykreslí hlášku o půjčení a ajaxem pošle ID filmu php skriptu.

Kapitola 3

Závěr

3.1 Zhodnocení práce

Webová aplikace splňuje všechny požadavky zadání, ale v této verzi nemůže být použita v reálném nasazení. Slouží pouze jako simulace půjčování. Pro reálné nasazení bych musel dodělat některé moduly, abych splňoval podmínky obchodního práva. Dále bych zde doplnil podrobnější informace o filmech. Například po kliknutí na film se zobrazí javascriptové okno s krátkým popisem, trailerem a s více fotkami. V administrativní části je zapotřebí udělat lepší úpravu informací u již vytvořeného filmu. Po dodělání těchto maličkostí by mohla být aplikace předvedena zaměstnavateli.